CLAIM AMENDMENT

Kindly Amend as follows:

CLAIMS

1. (Currently amended) A programming interface for generating a program embodied on one or more computer readable media, comprising instructions that, when implemented by a computing system, cause the computing system to generate:

a first group of services ~~related to~~ for generating graphical components to be included in an application being programmed;

a second group of services ~~related to~~ for binding properties of a class to a data source within the application being programmed; and

a third group of services ~~related to~~ for formatting content within the application being programmed.

2. (Original) A programming interface as recited in claim 1, wherein the first group of services includes a service that determines an appearance of the graphical components.

3. (Original) A programming interface as recited in claim 1, wherein the first group of services includes a service that determines a behavior of the graphical components.

4. (Original) A programming interface as recited in claim 1, wherein the first group of services includes a service that determines an arrangement of the graphical components.

5. (Original) A programming interface as recited in claim 1, wherein the first group of services includes a plurality of nested primitive controls that define the graphical components.

6. (Original) A programming interface as recited in claim 1, wherein the graphical components are defined by vector graphics.

7. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for animating at least one graphical component.

8. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for creating applications having navigation capabilities.

9. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for supporting electronic ink processing systems.

10. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for combining a plurality of different media types.

11. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for executing applications on a client using a browser-type interface.

12. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for automatically installing and executing an application.

13. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for serializing content.

14. (Currently amended) A programming interface as recited in claim 1, further comprising a fourth group of services ~~related to~~ for automating the generation of a user interface.

15. (Original) A software architecture comprising the programming interface as recited in claim 1.

16. (Currently amended) A programming interface embodied on one or more computer readable media, comprising instructions that when implemented cause a computing system to generate:

a first group of programming services ~~related to~~ for formatting content, for inclusion in an application being created, prior to displaying the content;

a second group of programming services ~~related to~~ for binding properties of a class, in the created application, to a data source; and

a third group of programming services ~~related to~~ for generating imaging effects for inclusion in the application.

17. (Currently amended) A programming interface as recited in claim 16, wherein the first group of programming services includes arranging a plurality of data elements.

18. (Currently amended) A programming interface as recited in claim 16, wherein the third group of programming services includes animating at least one graphical item.

19. (Currently amended) A programming interface as recited in claim 16, further comprising a fourth group of programming services ~~related to~~ for ~~creating applications~~ inclusion in the application that allow a user of the application to navigate between a plurality of images.

20. (Currently amended) A programming interface as recited in claim 16, further comprising a fourth group of programming services ~~related to~~ for editing previously created content included in the application being created.

21. (Currently amended) A programming interface as recited in claim 16, further comprising a fourth group of programming services ~~related to~~ for managing input received from an input device.

22. (Currently amended) A programming interface as recited in claim 16, further comprising a fourth group of programming services ~~related to~~ for enabling interoperability with other computing systems.

23. (Currently amended) A computer system including one or more microprocessors and one or more software programs, the one or more software programs utilizing an application program interface to request services from an operating system, the application program interface including separate commands to request services comprising the following groups of services:

a first group of services ~~related to~~ for generating graphical objects for use in an application being created;

a second group of services ~~related to~~ for creating components of the graphical objects for inclusion in the created application; and

a third group of services ~~related to~~ for modifying an appearance of the graphical objects.

24. (Currently amended) A computer system as recited in claim 23, wherein the first group of services includes a service for defining a behavior of at least one graphical object in a graphical user interface included in the created application.

25. (Original) A computer system as recited in claim 23, wherein the first group of services includes a service for defining arrangement of the graphical objects.

26. (Original) A computer system as recited in claim 23, wherein modifying an appearance of the graphical objects includes animating the graphical objects.

27. (Original) A computer system as recited in claim 23, wherein the second group of services includes services to generate geometric shapes.

28. (Currently amended) A computer system as recited in claim 23, wherein the application program interface further includes a fourth group of services ~~related to~~ for formatting text.

29. (Currently amended) A method for programming an application comprising:

calling one or more first functions ~~to facilitate~~ for formatting data for inclusion in an application;

calling one or more second functions ~~to facilitate~~ for creating graphical objects within the application; and

calling one or more third functions ~~to facilitate~~ for changing an appearance of the graphical objects within the application.

30. (Currently amended) A method as recited in claim 29, further including calling one or more fourth functions ~~to facilitate~~ for generating a user interface using a plurality of graphical objects.

31. (Currently amended) A method as recited in claim 29, further including calling one or more fourth functions ~~to facilitate~~ for creating runtime creation of a user interface.

32. (Currently amended) A method as recited in claim 29, further including:

calling one or more fourth functions ~~to facilitate~~ for generating a user interface using a plurality of graphical objects to be included in the application; and

calling one or more fifth functions ~~to facilitate~~ for runtime creation of the user interface.

33. (Original) A method as recited in claim 29, wherein the first functions facilitate:

receiving user input; and

arranging data elements on a display.

34. (Currently amended) A method as recited in claim 29, wherein the second functions ~~facilitate~~ are configured for generating geometric shapes.

35. (Currently amended) A method as recited in claim 29, wherein the second functions ~~facilitate~~ are configured for generating at least one geometric shape and the third functions ~~facilitate~~ are configured for modifying an appearance of the geometric shape.

36. (Currently amended) A system comprising:

means for exposing a first set of functions that enable creating a plurality of geometric shapes, such that the plurality of geometric shapes are configured for inclusion in an application being generated;

means for exposing a second set of functions that enable changing the manner in which the geometric shapes are arranged; and

means for exposing a third set of functions that enable modifying appearances of the geometric shapes.

37. (Original) A system as recited in claim 36, wherein the second set of functions further enable arrangement of the geometric shapes on a page to be rendered.

38. (Original) A system as recited in claim 36, wherein the plurality of geometric shapes include a line.

39. (Original) A system as recited in claim 36, wherein the third set of functions further enable associating imaging effects with at least one geometric shape.

40. (Original) A system as recited in claim 36, wherein the third set of functions further enable changing an appearance of a particular geometric shape over a period of time.

41. (Original) A system as recited in claim 36, further comprising means for exposing a fourth set of functions that enable generation of a user interface using the plurality of geometric shapes.

42. (Original) A system as recited in claim 36, further comprising means for exposing a fourth set of functions that enable associating a graphical object with one or more data sources.

43. (Original) A system as recited in claim 36, further comprising means for exposing a fourth set of functions that enable displaying data-specific versions of graphical objects.

44. (Currently amended) A method comprising:

calling one or more first functions ~~to facilitate~~ for creating components of graphical objects;

calling one or more second functions ~~to facilitate~~ for generating graphical objects for inclusion in an application being programmed;

calling one or more third functions ~~to facilitate~~ for modifying an appearance of the graphical objects in a display being created for use with the application;

calling one or more fourth functions ~~to facilitate~~ for arranging the graphical objects; and

calling one or more fifth functions ~~to facilitate~~ for associating the graphical objects with data sources.

45. (Currently amended) A method as recited in claim 44, further comprising calling one or more sixth functions ~~to facilitate~~ for navigating between ~~a plurality of~~ displays of content.

46. (Original) A method as recited in claim 44, wherein the components of the graphical objects include a plurality of shapes.

47. Canceled.

48. (Original) A method as recited in claim 44, wherein the third functions include functions that modify the appearance of a particular graphical object.

49. (Original) A method as recited in claim 44, wherein the third functions include functions that modify the appearance of one or more components of a graphical object.

50. (Currently amended) A method as recited in claim 44, wherein the third functions include functions that move graphical objects to different positions on the [[a]] display.

51. (Original) A method as recited in claim 44, wherein the third functions modify an appearance of a graphical object in response to user input.

52. (Original) A method as recited in claim 44, wherein the fourth functions modify an arrangement of graphical objects in response to user input.

53. (Withdrawn) A method of developing a program, the method comprising:

accessing a first group of functions to select components of a graphical image;

accessing a second group of functions to generate the graphical image;

accessing a third group of functions to modify an appearance of at least one component of the graphical image; and

accessing a fourth group of functions to format content associated with the graphical image.

54. (Withdrawn) A method as recited in claim 53, wherein the graphical image is a user interface.

55. (Withdrawn) A method as recited in claim 53, wherein the content is textual information.

56. (Withdrawn) A method as recited in claim 53, wherein the content is graphical data.

57. (Withdrawn) A method as recited in claim 53, further comprising accessing a fifth group of functions to associate a component of the graphical image with a data source.

58. (Withdrawn) A method as recited in claim 53, wherein the third group of functions further modify a behavior associated with the at least one component of the graphical image.

59. (New) A programming interface as recited in claim 1, wherein the application is configured to generate a graphical user interface to display graphical components in a selected format.

60. (New) A programming interface as recited in claim 16, wherein a resultant application is configured to generate a graphical user interface including graphical components.